



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

11/2

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/724,629	11/28/2000	Alex Truong	081862.P184	7864
7590	02/24/2006			EXAMINER
Sanjeet K. Dutta BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP Seventh Floor 12400 Wilshire Boulevard Los Angeles, CA 90025-1026			REILLY, SEAN M	
			ART UNIT	PAPER NUMBER
			2153	
DATE MAILED: 02/24/2006				

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/724,629	TRUONG ET AL.
	Examiner	Art Unit
	Sean Reilly	2153

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 07 November 2005.

2a) This action is FINAL. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-3, 5-24 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-3 and 5-24 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a) All b) Some * c) None of:

1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____ .	6) <input type="checkbox"/> Other: _____ .

DETAILED ACTION

This Office action is in response to Applicant's amendments and request for reconsideration filed on November 7, 2005. Claims 1-3 and 5-24 are presented for further examination. All independent claims have been amended.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 1-3 and 5-24 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

With regard to claims 1-3, 6, 11-14, 16-19, 21-24, the term "substantially all" renders each claim indefinite since it is not clear what criteria is used to render a group of commands as "substantially all commands." Thus, a person of skill in the art would not be able to ascertain the metes and bounds of the claimed invention. It is presumed at least one command required by each of the applications is loaded.

With regard to claims 7, 15, and 20, the language of these claims is cumbersome and should be rewritten. For example with regard to the limitation "transitioning the application from the no provisioning state to a quiescent state during which memories of the active card and the standby card are synchronized during the quiescent state," it is not clear if Applicant attempts to claim synchronizing cards: 1) during only the quiescent state, or 2) during only the transitioning from the no provisioning state to a quiescent state or 3) during both the act of

transitioning and the quiescent state (i.e. during both points one and two). Point number one is presumed.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

1. **Claims 1-3 and 5-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Muller et al. (U.S. Patent No. 6,650,640, hereinafter “Muller”), in view of Kicklighter (U.S. Patent No. 6,005,841).**

In considering claim 1, Muller discloses in a digital communications network, a method for controlling tasks performed on network cards (“NICs”) comprising:

Controlling applications executed within the network (col. 11, lines 23-25, wherein the applications are receiving packets and transferring packets), wherein controlling the applications comprises:

Transitioning each of the applications between one of a plurality of active states and one of a plurality of standby states (col. 11, lines 23-67; col. 17, lines 54-63, describing various active and standby states – i.e. extraction state 136 and wait state 400), wherein the plurality of active states comprise an active ready state (“state 132” in which a packet is received), a

quiescent state (“state 136” in which the packet does not move, but is merely analyzed), and a no-provisioning state (“state 130” start state – initialization of the card).

Wherein substantially all commands required by each of the applications are loaded into a memory of the active card for executing each of the applications during the active ready state (Col 11, lines 22-26 and Col 17, lines 47-53, initialization of the NIC card).

However Muller does not explicitly disclose that the states occur on separate cards, one being an active card and the other being a standby card. Nonetheless, transitioning application states between separate active and standby cards, rather than using a single card as taught by Muller, is well known, as evidenced by Kicklighter. In a similar art, Kicklighter discloses a system for controlling states in cards of a switching system (Abstract), wherein the states of applications change between active and standby states (col. 7, lines 5-11), and wherein the states occur on separate active cards and standby cards (col. 6, lines 32-47, describing “active card 38(A)” and “standby card 38(S)”). Kicklighter further discloses the memories of the active card and the standby card are synchronized during the quiescent state (i.e. during all operating states of the active card, the standby card is in listening mode and is always synchronized with the active card, see *inter alia*, Col 2, lines 5-44 and Col 8, lines 17-32). Given the teaching of Kicklighter, it would have been obvious to a person having ordinary skill in the art to separate the single card system of Muller into two separate cards, in order to increase the speed of the system by requiring faster and simpler processing on each card.

With further regard to the limitation “wherein at least a portion of network management requests for configuring the active network card are rejected during the no-provisioning state,” Kicklighter disclosed loading the line card switch program into memory for execution (Col 5,

lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running.

In considering claim 2, Muller further discloses that an application state machine controls the execution of the application (cols. 11-12, wherein a state machine necessarily controls the application which is transitioning through various states).

In considering claim 3, Muller further discloses receiving control messages from a shelf manager (“SYM process”), and communicating via APIs to the application, wherein the shelf manager may be located on a remote network card (“SYM process” responsible for the management of all system resources Col 5, lines 37-47).

In considering claim 5, Muller further discloses that the standby states comprise:
A standby locked state (“state 430,” col. 18, lines 56-67, wherein if the packet does not match the desired criteria, it is not actively processed and the process ends).

In considering claim 6, Muller discloses in a digital communication network, a method for controlling tasks performed on network cards (“NICs,” col. 11, lines 23-26), comprising:
Switching the state of an application in an active state to a standby state, comprising,

Transitioning the application from the active state to a quiescent state (col. 11, lines 27-59, wherein the application transitions from “state 132” in which a packet is actively received, to “state 136,” wherein the packet does not move but is merely analyzed); and

Transitioning the application from the quiescent state to the standby state (col. 17, lines 54-63, in which the card returns to the “wait state” when packets are not being processed, and thus transitions from the “state 136” described above to the “wait state” after the packet is fully processed).

Wherein substantially all commands required by each of the applications are loaded into a memory of the active card for executing each of the applications during the active ready state (Col 11, lines 22-26 and Col 17, lines 47-53, initialization of the NIC card).

However Muller does not explicitly disclose that the states occur on separate cards, one being an active card and the other being a standby card. Nonetheless, transitioning application states between separate active and standby cards, rather than using a single card as taught by Muller, is well known, as evidenced by Kicklighter. In a similar art, Kicklighter discloses a system for controlling states in cards of a switching system (Abstract), wherein the states of applications change between active and standby states (col. 7, lines 5-11), and wherein the states occur on separate active cards and standby cards (col. 6, lines 32-47, describing “active card 38(A)” and “standby card 38(S)”). Kicklighter further discloses the memories of the active card and the standby card are synchronized during the quiescent state (i.e. during all operating states of the active card, the standby card is in listening mode and is always synchronized with the active card, see inter alia, Col 2, lines 5-44 and Col 8, lines 17-32). Given the teaching of Kicklighter, it would have been obvious to a person having ordinary skill in the art to separate

the single card system of Muller into two separate cards, in order to increase the speed of the system by requiring faster and simpler processing on each card.

With further regard to the limitation “wherein at least a portion of network management requests for configuring the active network card are rejected during the no-provisioning state,” Kicklighter disclosed loading the line card switch program into memory for execution (Col 5, lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running.

In considering claim 7, Muller discloses in a digital communication network, a method for controlling tasks performed on network cards (“NICs,” col. 11, lines 23-26), comprising:

Switching the state of an application in an active state to a standby locked state, comprising,

Transitioning the application from the active state to a no-provisioning state (col. 11, lines 27-59; wherein the application transitions from “state 132” in which a packet is actively received, to “state 130” in which no provisioning occurs during reboot of the NIC);

Transitioning the application from the no provisioning state to a quiescent state (col. 12, lines 15-18-27, wherein the application transitions from “state 130” to “state 144,” wherein the packet is not transferred to the host memory, and thus is quiescent); and

Transitioning the application from the quiescent state to the standby locked state (col. 17, lines 54-63, in which the card returns to the “wait state” when packets are not being processed,

and thus transitions from the “state 144” described above to the “wait state” after the packet is fully processed).

However Muller does not explicitly disclose that the states occur on separate cards, one being an active card and the other being a standby card. Nonetheless, transitioning application states between separate active and standby cards, rather than using a single card as taught by Muller, is well known, as evidenced by Kicklighter. In a similar art, Kicklighter discloses a system for controlling states in cards of a switching system (Abstract), wherein the states of applications change between active and standby states (col. 7, lines 5-11), and wherein the states occur on separate active cards and standby cards (col. 6, lines 32-47, describing “active card 38(A)” and “standby card 38(S)”). The memories of the active card and the standby card are synchronized during the quiescent state (i.e. during all operating states of the active card, the standby card is in listening mode and is always synchronized with the active card, see *inter alia*, Col 2, lines 5-44 and Col 8, lines 17-32). Additionally the standby card is in a ready state but does not communicate with the corresponding application of the active card (i.e. the standby card is in listen mode and ready to perform the duties of the active card but does not communicate with the active card, Col 16-33). Given the teaching of Kicklighter, it would have been obvious to a person having ordinary skill in the art to separate the single card system of Muller into two separate cards, in order to increase the speed of the system by requiring faster and simpler processing on each card.

With further regard to the limitation “wherein at least a portion of network management requests for configuring the active network card are rejected during the no-provisioning state,” Kicklighter disclosed loading the line card switch program into memory for execution (Col 5,

lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running.

In considering claim 8, Muller further discloses that the standby locked state does not allow disk database access nor access to write to RAM (i.e. the “idle” or “wait” state only allows the card to listen for packets).

In considering claim 9, Muller further discloses that the no provisioning state does not allow access to write to a disk database (i.e. “state 142” only waits and does not allow writing to a database).

In considering claim 10, Muller further discloses that the quiescent state does not allow access to write to a disk database nor access to write to RAM (i.e. “state 144” only determines if the packet will soon be transferred, and does not allow write access to a database or RAM).

In considering claim 11, Muller discloses in a digital communications network, a method for controlling tasks performed on network cards (“NICs”), comprising:

Upgrading code of an application in a standby state to an active state, comprising:
Transitioning the application from the standby state to a no provisioning state (col. 17, lines 54-63; col. 12, lines 10-20, i.e. from the “wait state” to “state 142” in which no provisioning occurs); and

Transitioning the application from the no provisioning state to the active state (col. 12, lines 20-31, wherein the application transitions from “state 142” to “state 146” in which the packet is transferred).

Wherein substantially all commands required by each of the applications are loaded into a memory of the active card for executing each of the applications during the active ready state (Col 11, lines 22-26 and Col 17, lines 47-53, initialization of the NIC card).

However Muller does not explicitly disclose that the states occur on separate cards, one being an active card and the other being a standby card. Nonetheless, transitioning application states between separate active and standby cards, rather than using a single card as taught by Muller, is well known, as evidenced by Kicklighter. In a similar art, Kicklighter discloses a system for controlling states in cards of a switching system (Abstract), wherein the states of applications change between active and standby states (col. 7, lines 5-11), and wherein the states occur on separate active cards and standby cards (col. 6, lines 32-47, describing “active card 38(A)” and “standby card 38(S)”). Kicklighter further discloses the memories of the active card and the standby card are synchronized during the quiescent state (i.e. during all operating states of the active card, the standby card is in listening mode and is always synchronized with the active card, see *inter alia*, Col 2, lines 5-44 and Col 8, lines 17-32). Given the teaching of Kicklighter, it would have been obvious to a person having ordinary skill in the art to separate the single card system of Muller into two separate cards, in order to increase the speed of the system by requiring faster and simpler processing on each card.

With further regard to the limitation “wherein at least a portion of network management requests for configuring the active network card are rejected during the no-provisioning state,”

Kicklighter disclosed loading the line card switch program into memory for execution (Col 5, lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running.

In considering claim 12, claim 12 presents a system for performing the method described in claim 1, and is thus rejected for the same reasons.

In considering claim 13, claim 13 presents a system for performing the method described in claim 3, and is thus rejected for the same reasons.

In considering claim 14, claim 14 presents a system for performing the method described in claim 6, and is thus rejected for the same reasons.

In considering claim 15, claim 15 presents a system for performing the method described in claim 7, and is thus rejected for the same reasons.

In considering claim 16, claim 16 presents a system for performing the method described in claim 11, and is thus rejected for the same reasons.

In considering claim 17, claim 17 presents a computer readable medium for performing the method described in claim 1, and is thus rejected for the same reasons.

In considering claim 18, claim 18 presents a computer readable medium for performing the method described in claim 3, and is thus rejected for the same reasons.

In considering claim 19, claim 19 presents a computer readable medium for performing the method described in claim 6, and is thus rejected for the same reasons.

In considering claim 20, claim 20 presents a computer readable medium for performing the method described in claim 7, and is thus rejected for the same reasons.

In considering claim 21, claim 21 presents a computer readable medium for performing the method described in claim 11, and is thus rejected for the same reasons.

In considering claim 22, Muller discloses in a digital communications network, a system for controlling tasks performed on network cards (“NICs,” col. 11, lines 23-26) comprising:

A CPU subsystem (“CPU,” col. 53, lines 8-24);

One or more input/output ports connected to the CPU subsystem for communicating with the network (“Input port processing module 104,” Fig. 1); and

Special hardware connected to the CPU subsystem via a bus (“PCI bus,” col. 50, lines 15-23), wherein the CPU subsystem controls applications executed within the network (col. 11, lines 23-26, wherein the NIC is ultimately controlled by the host CPU), wherein the plurality of active states comprise an active ready state (“state 132” in which a packet is received), a quiescent state

(“state 136” in which the packet does not move, but is merely analyzed), and a no-provisioning state (“state 130” start state – initialization of the card)..

Wherein substantially all commands required by each of the applications are loaded into a memory of the active card for executing each of the applications during the active ready state (Col 11, lines 22-26 and Col 17, lines 47-53, initialization of the NIC card).

However Muller does not explicitly disclose that the states occur on separate cards, one being an active card and the other being a standby card. Nonetheless, transitioning application states between separate active and standby cards, rather than using a single card as taught by Muller, is well known, as evidenced by Kicklighter. In a similar art, Kicklighter discloses a system for controlling states in cards of a switching system (Abstract), wherein the states of applications change between active and standby states (col. 7, lines 5-11), and wherein the states occur on separate active cards and standby cards (col. 6, lines 32-47, describing “active card 38(A)” and “standby card 38(S)”). Kicklighter further discloses the memories of the active card and the standby card are synchronized during the quiescent state (i.e. during all operating states of the active card, the standby card is in listening mode and is always synchronized with the active card, see inter alia, Col 2, lines 5-44 and Col 8, lines 17-32). Given the teaching of Kicklighter, it would have been obvious to a person having ordinary skill in the art to separate the single card system of Muller into two separate cards, in order to increase the speed of the system by requiring faster and simpler processing on each card.

With further regard to the limitation “wherein at least a portion of network management requests for configuring the active network card are rejected during the no-provisioning state,” Kicklighter disclosed loading the line card switch program into memory for execution (Col 5,

lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running.

In considering claim 23, Muller further discloses a disk database (“flow database 110,” on the NIC, col. 8, lines 50-60) connected to the CPU system via a PCI bus (col. 50, lines 18-20, “NIC 100 is coupled to the host computer by a PCI bus”).

In considering claim 24, Muller further discloses that the CPU subsystem comprises:

- A central processing unit (“CPU”);
- A system controller connected to the central processing unit (“control queue 118,” col. 52, lines 43-67; Fig. 1);

Random access memory (“random access memory”) connected to the system controller (col. 52, lines 43-50); and

An application state machine for transitioning applications between one of a plurality of active states and one of a plurality of standby states (col. 11, lines 23-67; col. 17, lines 54-63, describing various active and standby states – i.e. extraction state 136 and wait state 400)

Response to Arguments

Applicant asserts that Muller and Kicklighter both failed to disclose various states including “the active ready state, the no provision state, a standby locked state, and a quiescent state.” Examiner respectfully disagrees with this assertion. As clearly indicated in the above

rejection these states are disclosed by at least Muller, i.e. the active ready state (state 132), the no provision state (state 130), the standby locked state (state 430), and the quiescent state (state 136). Additionally with further regard to the no provision state and rejecting “network management requests for configuring the active network card” during the no-provisioning state, Kicklighter disclosed a loading state where the line card switch program is loaded into memory for execution (Col 5, lines 34-54). While loading the line card switch program into memory as part of the line card boot initialization process, the line card switch program is unable to service any network management requests since the line card switch program is not running. Thus, network management requests are rejected during this loading state.

Applicant also asserts “there is no suggestion within the cited references to combine with each other.” Examiner respectively disagrees. Examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found in the references themselves. However motivation may also be found in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992).

Conclusion

1. The prior art made of record, in PTO-892 form, and not relied upon is considered pertinent to applicant's disclosure.
2. This office action is made **NON-FINAL**.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sean Reilly whose telephone number is 571-272-4228. The examiner can normally be reached on M-F 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Glen Burgess can be reached on 571-272-3949. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

February 16, 2006



GLENTON B. BURGESS
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100